# The cross-section of expected stock returns: A universal approach to anomaly discovery

Benjamin Moritz[1] Tom Zimmermann[2]

[1]Ludwig Maximilian University Munich
Chair of Financial Econometrics

[2]Harvard University
Department of Economics

November 8, 2013

# MOTIVATION

- What is the scope for machine learning in economics and finance?
- Which concepts and techniques can be transferred?

# THIS PROJECT: CROSS-SECTION OF STOCK RETURNS

Table: Literature surveys

| Author(s) | Return predictive signals (RPS) |
|---|---|
| Subrahmanyam (2010) | 50 |
| Harvey, Liu and Zhu (2013) | 185 |
| McLean and Pontiff (2013) | 82 |
| Green, Hang and Zhang (2013) | 330 |

> *[...] either US stock markets are pervasively inefficient, or there exist a much larger number of rationally priced sources of risk in equity returns than previously thought.*
> Green et al (2013)

# COCHRANE (2011, PRESIDENTIAL ADDRESS)

*We are going to have to repeat Fama and Frenchs anomaly digestion, but with many more dimensions. We have a lot of questions to answer:*
*First, which characteristics really provide independent information about average returns? Which are subsumed by others?*
*Second, does each new anomaly variable also correspond to a new factor formed on those same anomalies? [...]*
*Third, how many of these new factors are really important? [...]*

# THE CROSS-SECTION OF STOCK RETURNS

- Data-mining concerns
- Many factors might be correlated
- Portfolio sorts can only control for few dimensions
- ”Difficult to interpret the state of the literature” (Subrahmanyam (2010))
- Goal: Comprehensive study to analyze joint impact of signals

# TODAY: DISSECT MOMENTUM

- Usually modeled as stock return over past 12 months, skipping most recent month (M1-12)
- Novy-Marx (2012, JoFE) shows that an M7-12 strategy is more profitable than an M1-6 strategy
- Goyal and Wahal (2012) cannot find this effect in 36 out of 37 international markets
- What is going on? Which types of momentum can be learned from the data alone?

# OUTLINE

1. Lay out the general framework
   - Bound data-mining concerns
   - Control for correlated signals
2. Illustrate procedure using momentum
3. Zoom in on the debate about short-horizon return vs intermediate-horizon return

# PRELIMINARY FINDINGS FOR MOMENTUM

- Single performance evaluation: both SH and MH return look important
- Joint evaluation: SH return features can be learned from the data, MH return not
- Test: Precision of predictions decreases for exclusion of SH return, not for exclusion of MH return
- Consistent with Goyal and Wahal (2012)

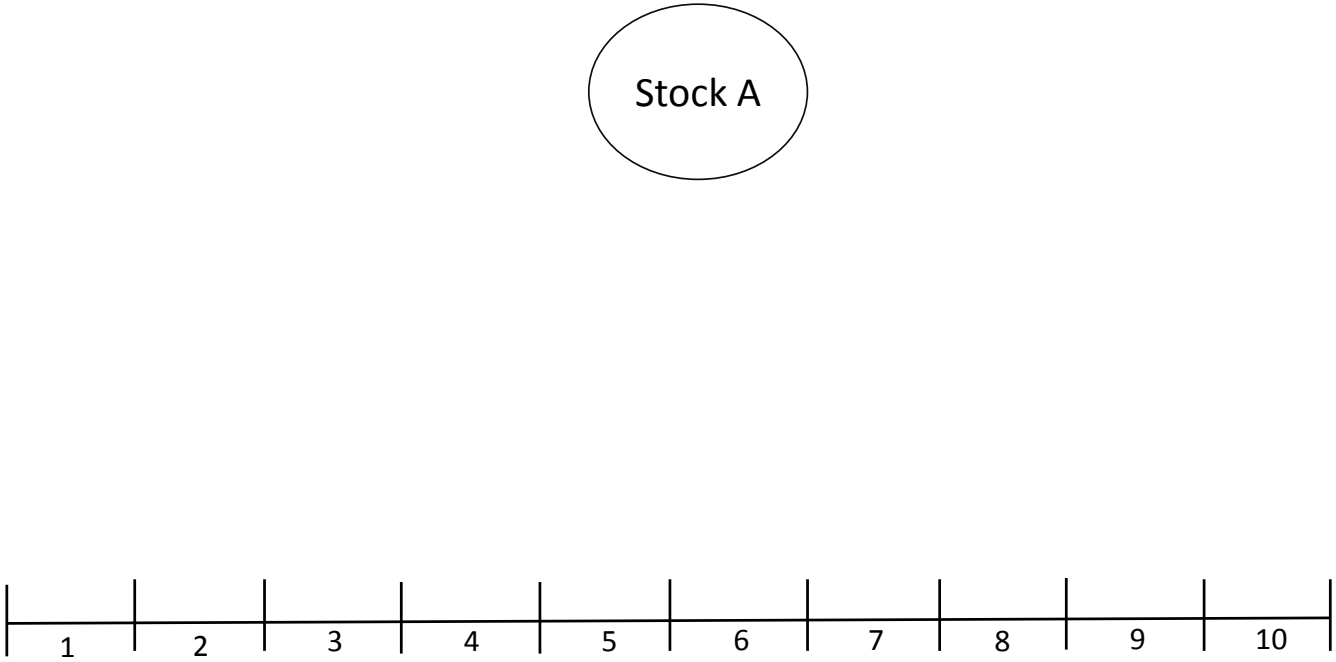
# SETUP

# ROADMAP

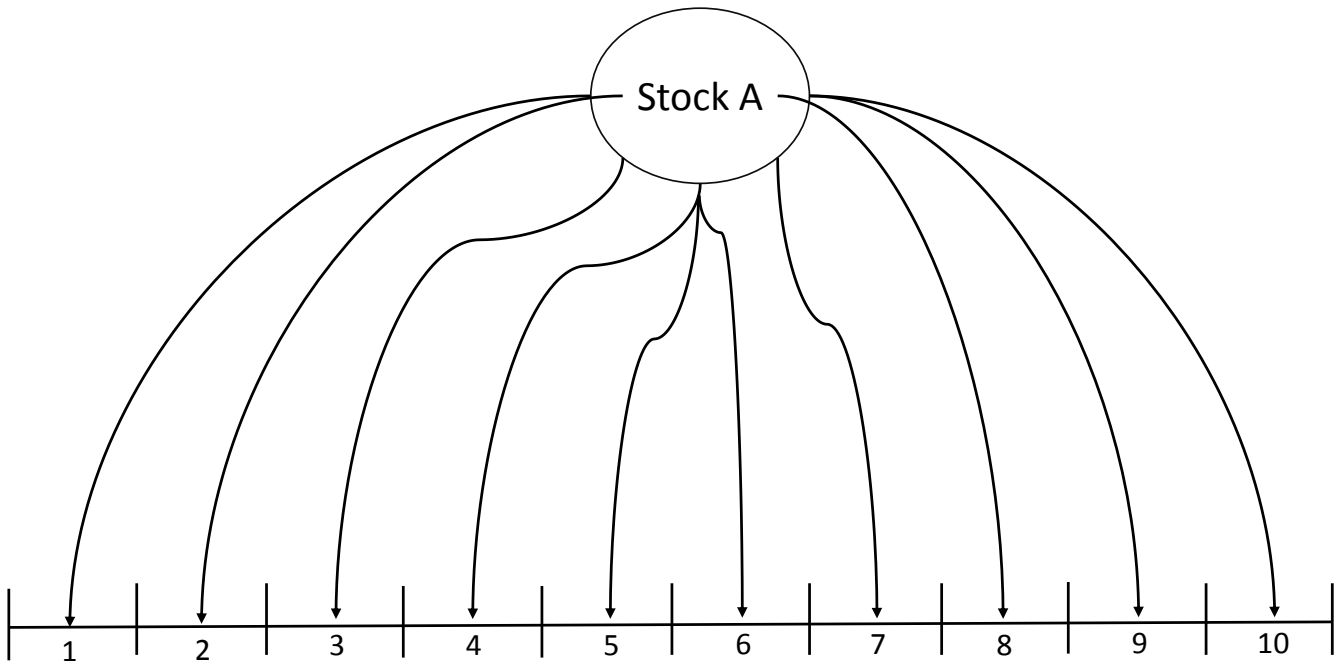
- Supervised learning usually starts from a prediction task
  - If a signal is relevant, it should be useful in that task

1. Define prediction task
2. Talk about how we solve it

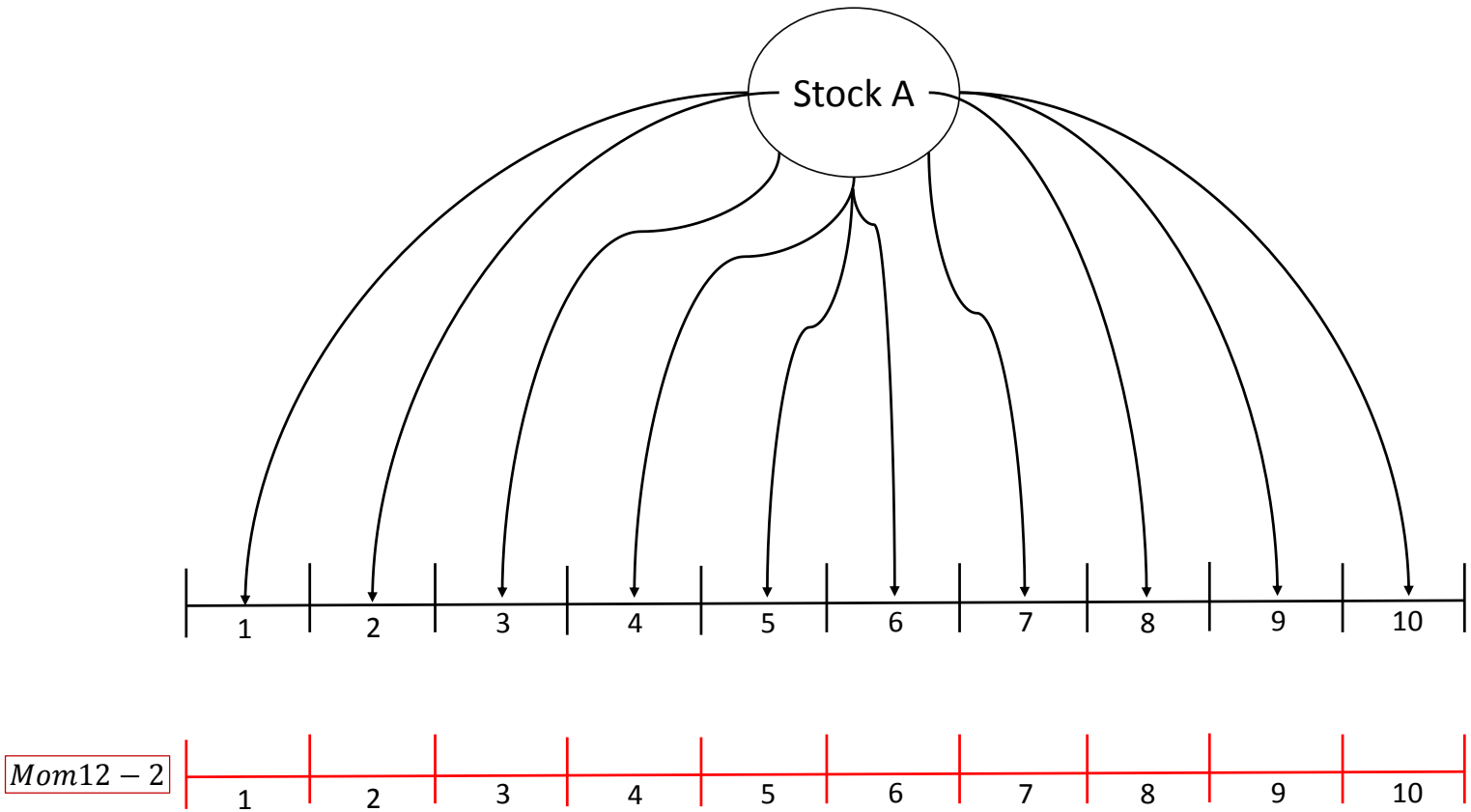
# PREDICTION TASK



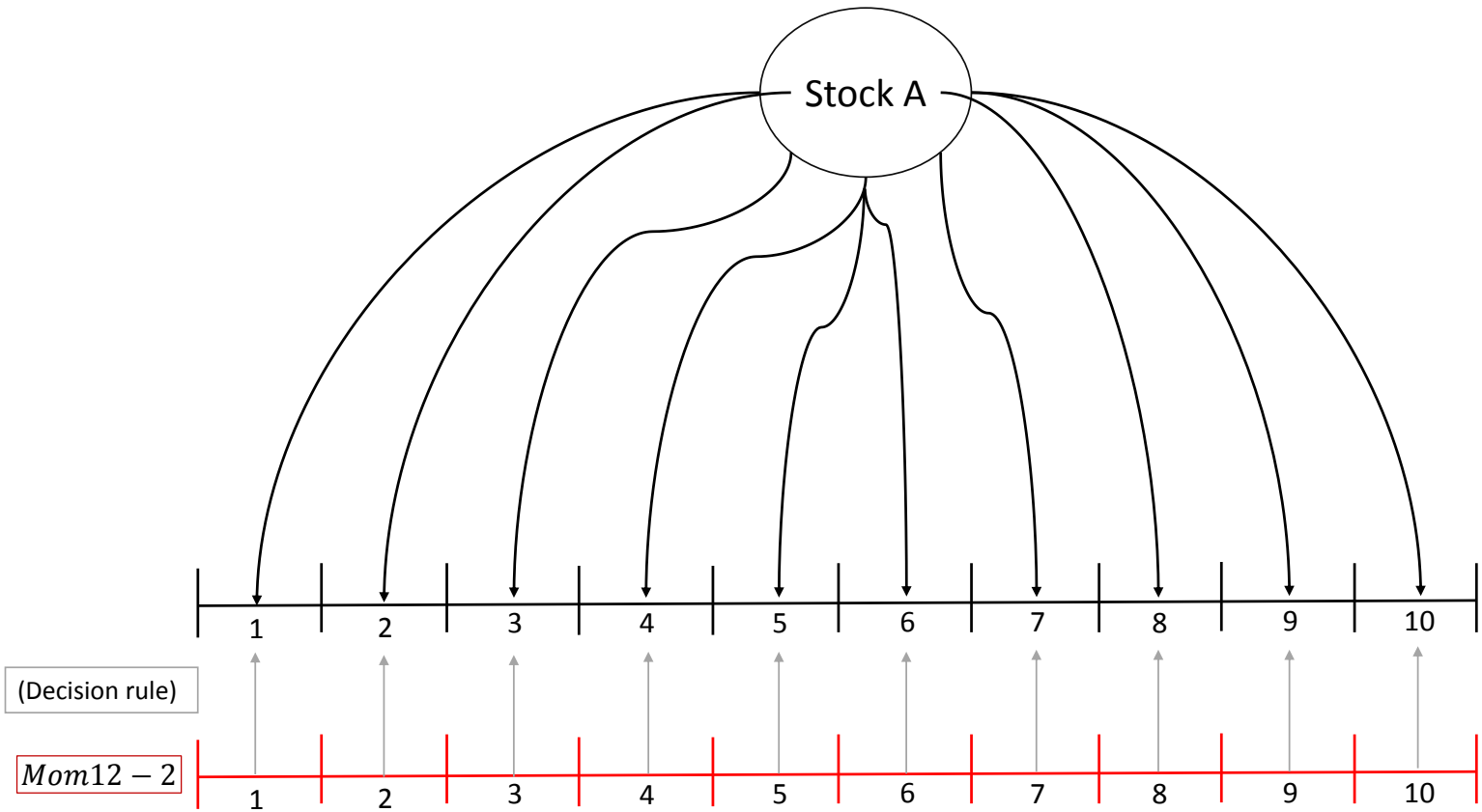
# PREDICTION TASK



# PREDICTION TASK



# PREDICTION TASK

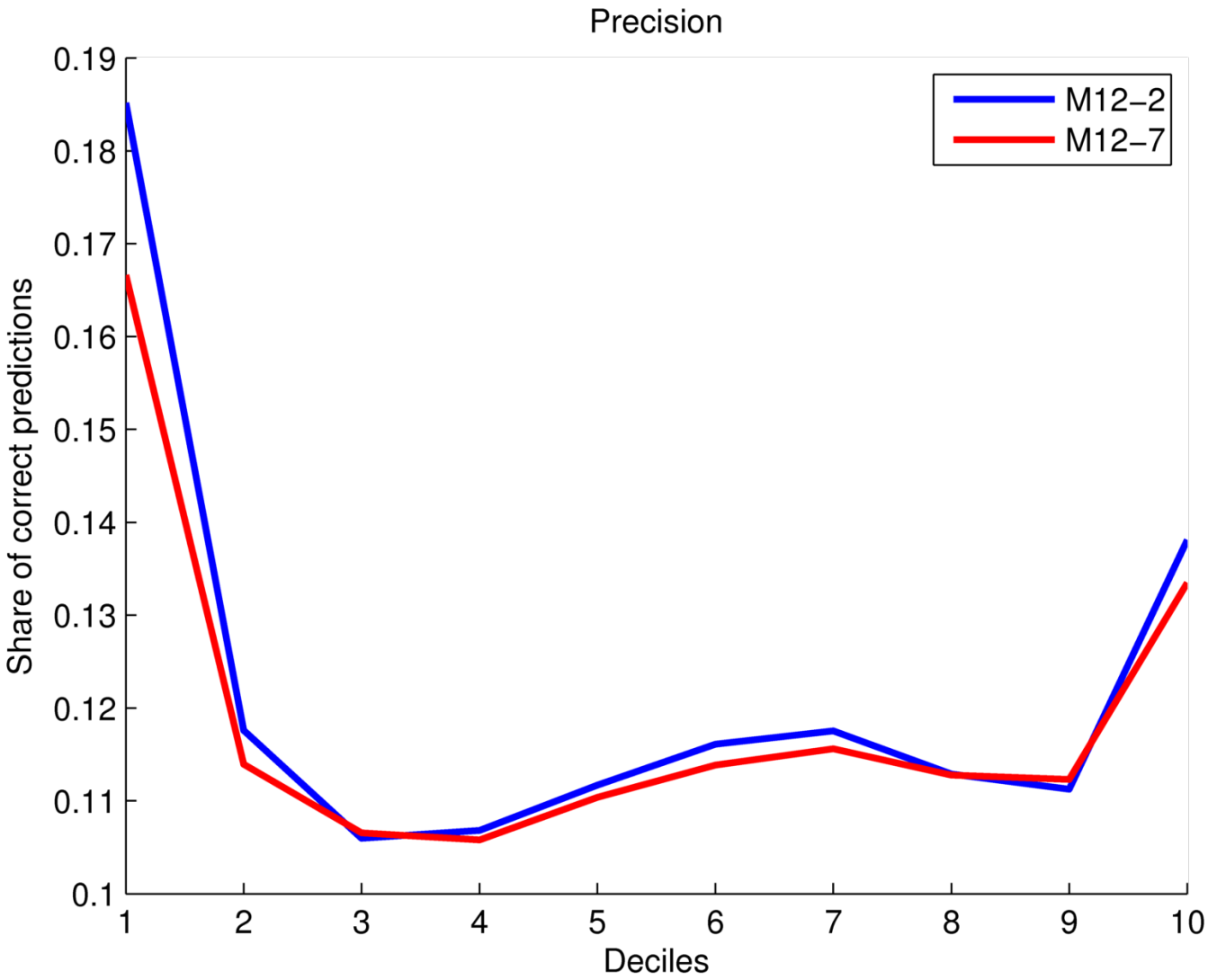


# PERFORMANCE MEASURE

We measure performance by *precision*:

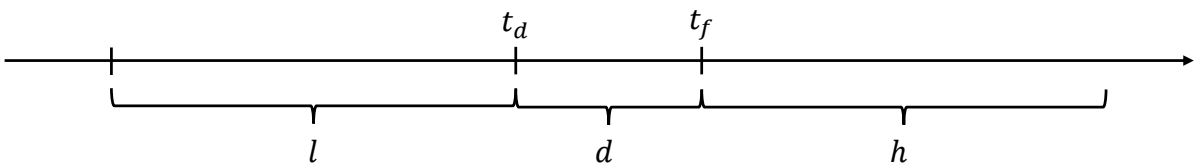
$$\text{Precision}(k) = \frac{\text{Correct predictions of class } k}{\text{All predictions of class } k}$$

# TWO MOMENTUM STRATEGIES ON FULL SAMPLE



# FEATURE CONSTRUCTION

- Construct momentum functions as $M(d, l)$
- $d \in \{0, \ldots, 6\}$
- $l \in \{1, \ldots, 18\}$
- $M(\cdot, \cdot)$ maps returns into quantiles $\rightarrow$ 126 features/strategies
- More generally: $M(d, l, \mathrm{h})$. Today: $h = 1$.

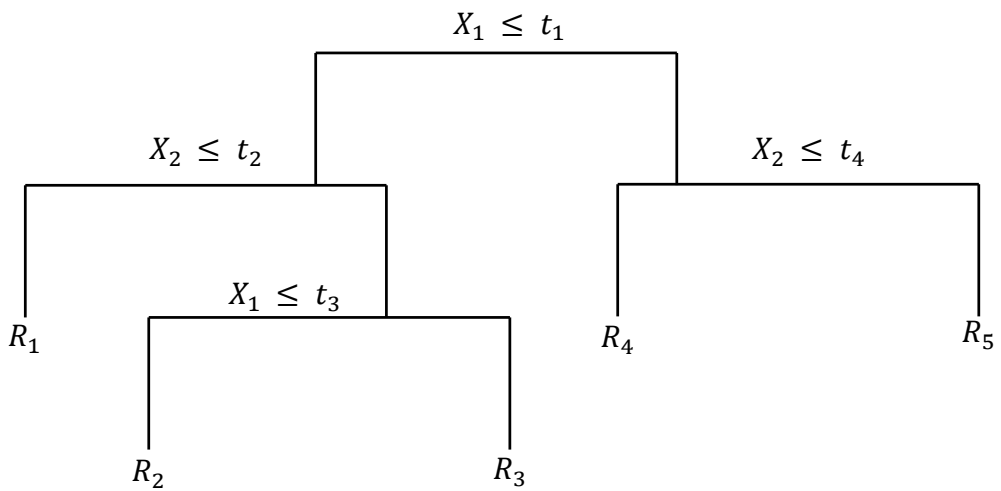


# ESTIMATION

1. Decision-tree algorithm to deal with correlation of features
2. Random forest to reduce variance
3. Hold-out sample to evaluate performance
4. Out-of-bag sampling to estimate *expected precision* and *feature importance* on test data

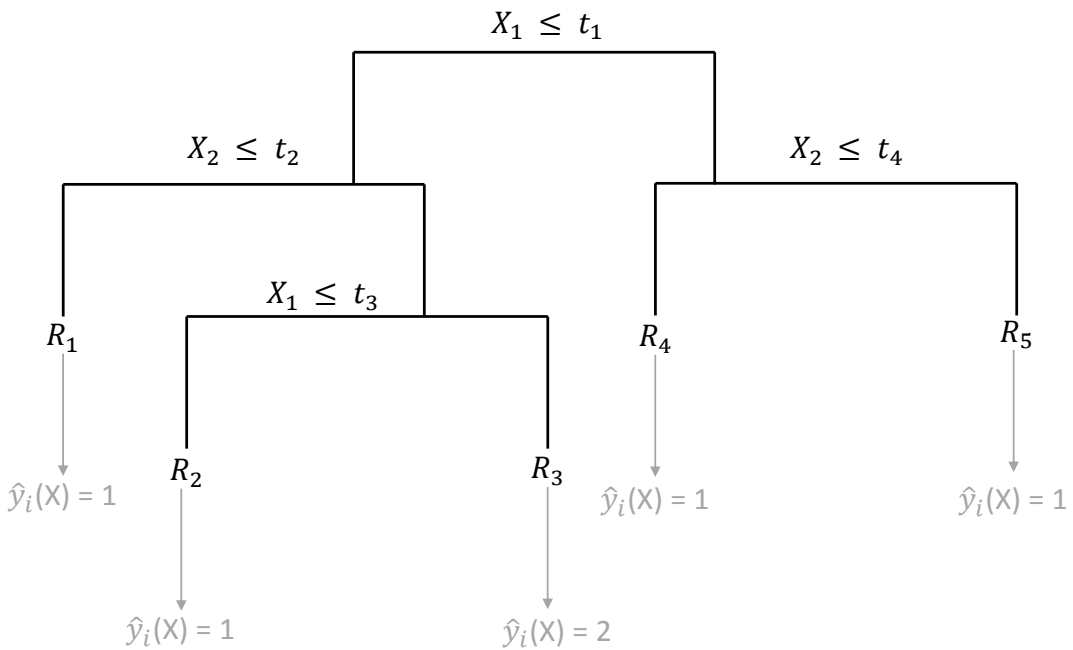
# DECISION-TREE ALGORITHM I

- Let $y_i$ be a categorical outcome variable, and $x_i$ a $p \times 1$ vector of candidate features.
- Basic idea with two features:



# DECISION-TREE ALGORITHM I

- Let $y_i$ be a categorical outcome variable, and $x_i$ a $p \times 1$ vector of candidate features.
- Basic idea with two features:



# DECISION-TREE ALGORITHM II

- Finding optimal data partitioning is NP-complete → Use greedy procedure
- Share of class $k$ in region $R_m$:

$$\hat{p}_{mk} = \frac{1}{N_m} \sum_{x_i \in R_m} I(y_i = k)$$

- At each node, define *node impurity* as

$$H(\hat{p}) = -\sum_{k=1}^{K} \hat{p}_{mk} log(\hat{p}_{mk})$$

- Find the feature $j$ and the corresponding threshold that minimize $H(\hat{p})$ at each node

# DECISION-TREE ALGORITHM III

- Automatic variable selection
- Robust to outliers
- Easily scalable to large $N$
- Prediction for new data point $x$ is $\hat{T}(x)$.
  - Problem: Single tree is often a *weak learner*
  - Predictive accuracy low $\rightarrow$ Random forests

# ESTIMATION

1. Decision-tree algorithm to deal with correlation of features
2. Random forest to reduce variance
3. Hold-out sample to evaluate performance
4. Out-of-bag sampling to estimate *expected precision* and *feature importance* on test data

# RANDOM FORESTS

Idea: Reduce variance of an estimate by averaging together many estimates

- Suppose $B$ samples, identical but not independently distributed. $\hat{T}_b(x)$ is prediction of weak learner $b$ at $x$. Let

$$\overline{\hat{T}_B(x)} = \frac{1}{B}\sum_{b=1}^{B}\hat{T}_b(x)$$

$$Var(\overline{\hat{T}_B(x)}) = \rho\sigma^2 + \frac{1-\rho}{B}\sigma^2 \to \rho\sigma^2, \text{for } B \to \infty$$

- Random forests: Reduce $\rho$ by training base learners on
  - Stochastic perturbation of the feature space
  - Random subset of the data

# RANDOM FORESTS: ALGORITHM

## Algorithm

*For* $b = 1 : B$

1. *Draw bootstrap sample* $Z$ *of size* $N$ *from training data*
2. *Grow tree* $T_b$ *that uses* $m < p$ *candidate features*
3. *Collect* $\{T_b\}_1^B$

*Class prediction is* $\hat{C}_{RF}^B(x) = \text{majority vote}\{\hat{C}_b(x)\}_1^B$

# RANDOM FORESTS: COMMENTS

- Bias of RF is the same as bias for any one tree $\rightarrow$ improvements in prediction are result of variance reduction alone
- Accuracy is often very high
- Random Forests lose the simple interpretability of a single tree

# ESTIMATION

1. Decision-tree algorithm to deal with correlation of features
2. Random forest to reduce variance
3. Hold-out sample to evaluate performance
4. Out-of-bag sampling to estimate *expected precision* and *feature importance* on test data

# ESTIMATION

1. Decision-tree algorithm to deal with correlation of features
2. Random forest to reduce variance
3. Hold-out sample to evaluate performance
4. Out-of-bag sampling to estimate *expected precision* and *feature importance* on test data

# ESTIMATION

1. Decision-tree algorithm to deal with correlation of features
2. Random forest to reduce variance
3. Hold-out sample to evaluate performance
4. Out-of-bag sampling to estimate *expected precision* and *feature importance* on test data

# RANDOM FORESTS: EVALUATION

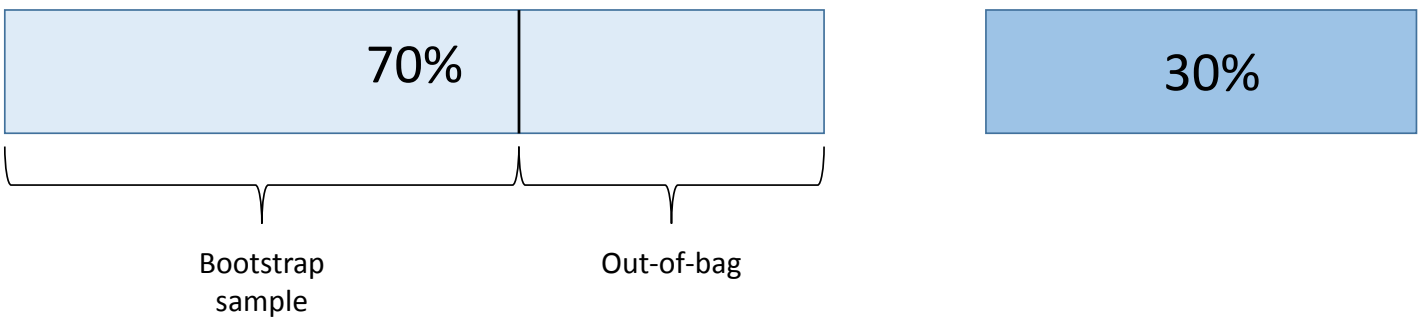
## Expected precision

- Bootstrap sampling with replacement leaves out $\approx 33$ % of observations for each tree.
- Use those to estimate precision and average over all trees
- Compare to precision on hold-out sample

# RANDOM FORESTS: EVALUATION

## Predictive precision

- Bootstrap sampling with replacement leaves out $\approx 33$ % of observations for each tree.
- Use those to estimate the expected precision and average over all trees
- Compare to precision on hold-out sample



# RANDOM FORESTS: EVALUATION

Feature importance

1. Compute prediction accuracy for OOB samples for $T_b$
2. Randomly permute values of variable $j$
3. Compute decrease in accuracy
4. Average over all trees

# EMPIRICS

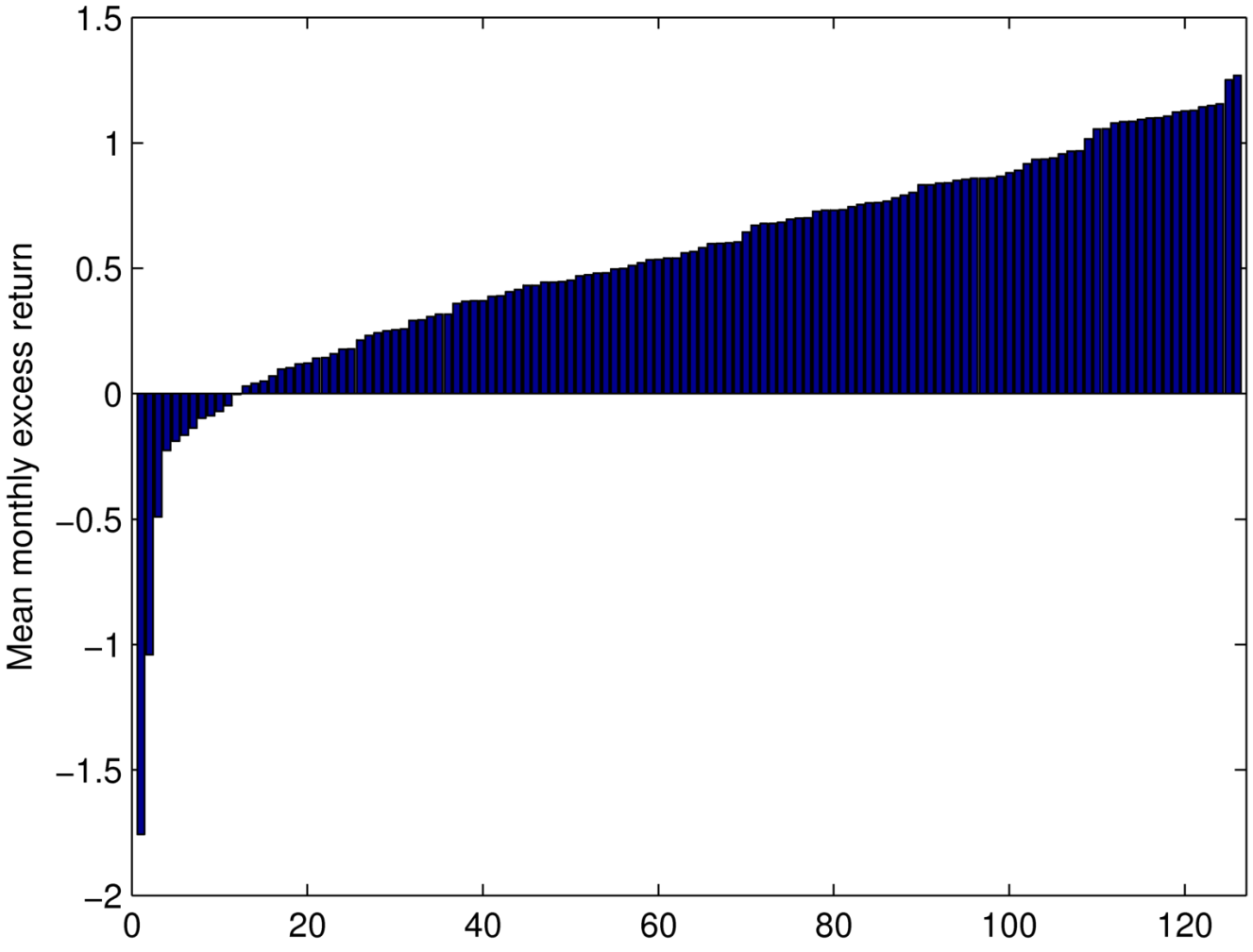
# DATA

- CRSP monthly stock file, 1926-2012
- Exclude penny stocks
- Ordinary common shares only
- Truncate .5% and 99.5% monthly return percentile
- FF factors and UMD

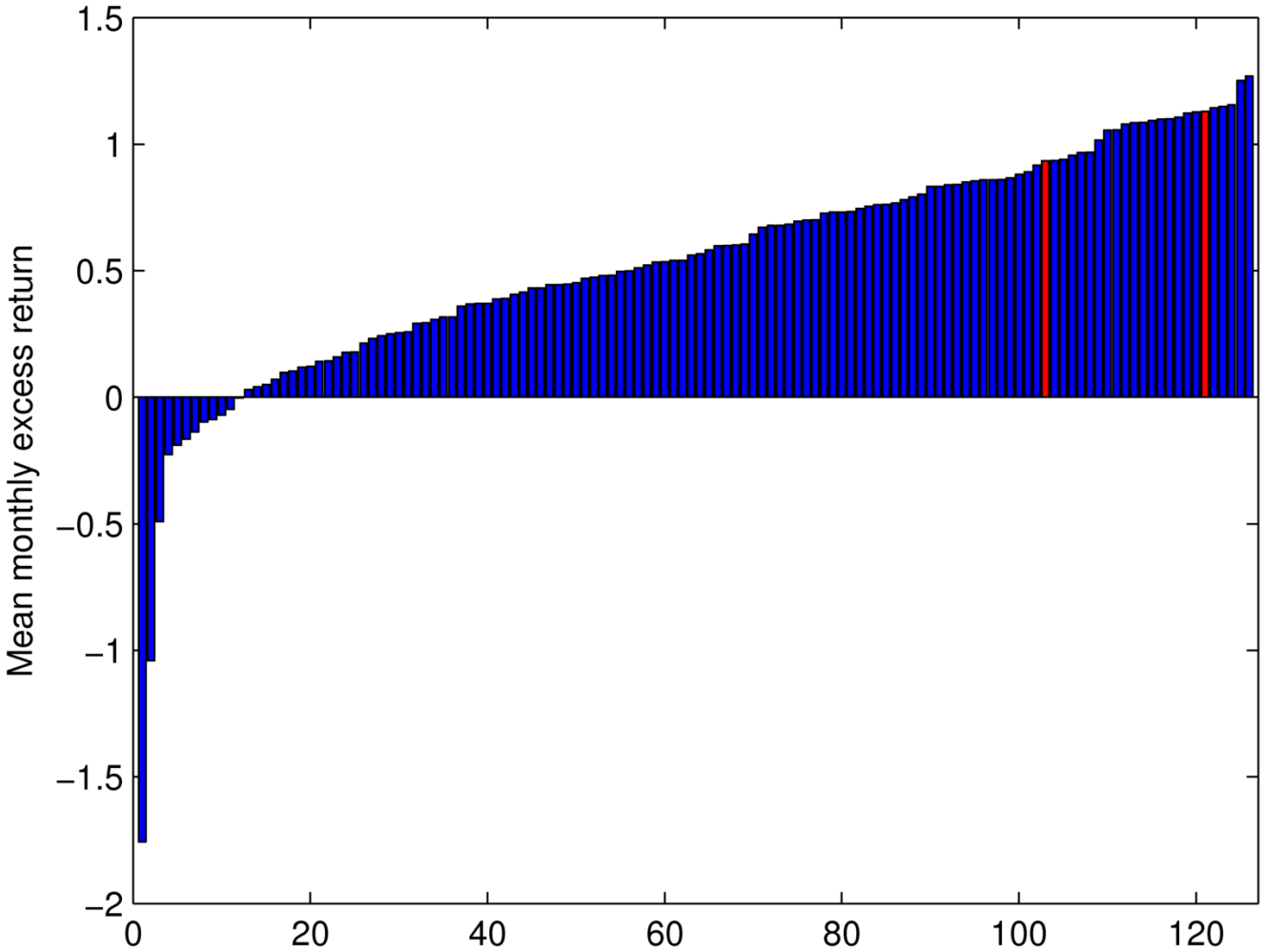
# IMPLEMENTATION

- Construct 126 predictors $M(d, l)$
- Compute decile sorting for each predictor and one-month future return
- Tuning parameters (tree depth, number of randomly chosen predictors,...) can be cross-validated (not done yet)
- One RF run takes 6-7 hours with parallel processing

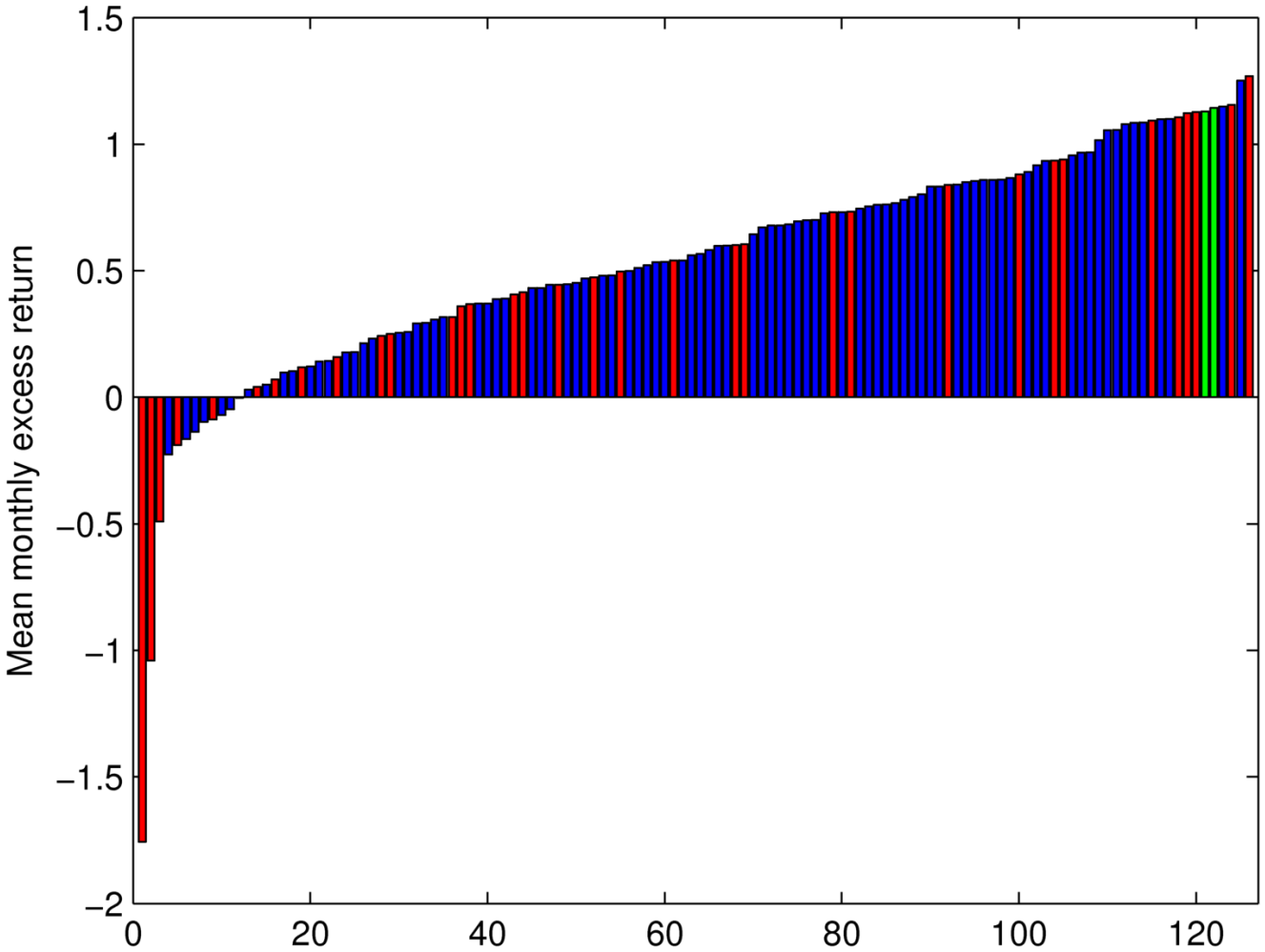
# SINGLE FEATURE PERFORMANCE



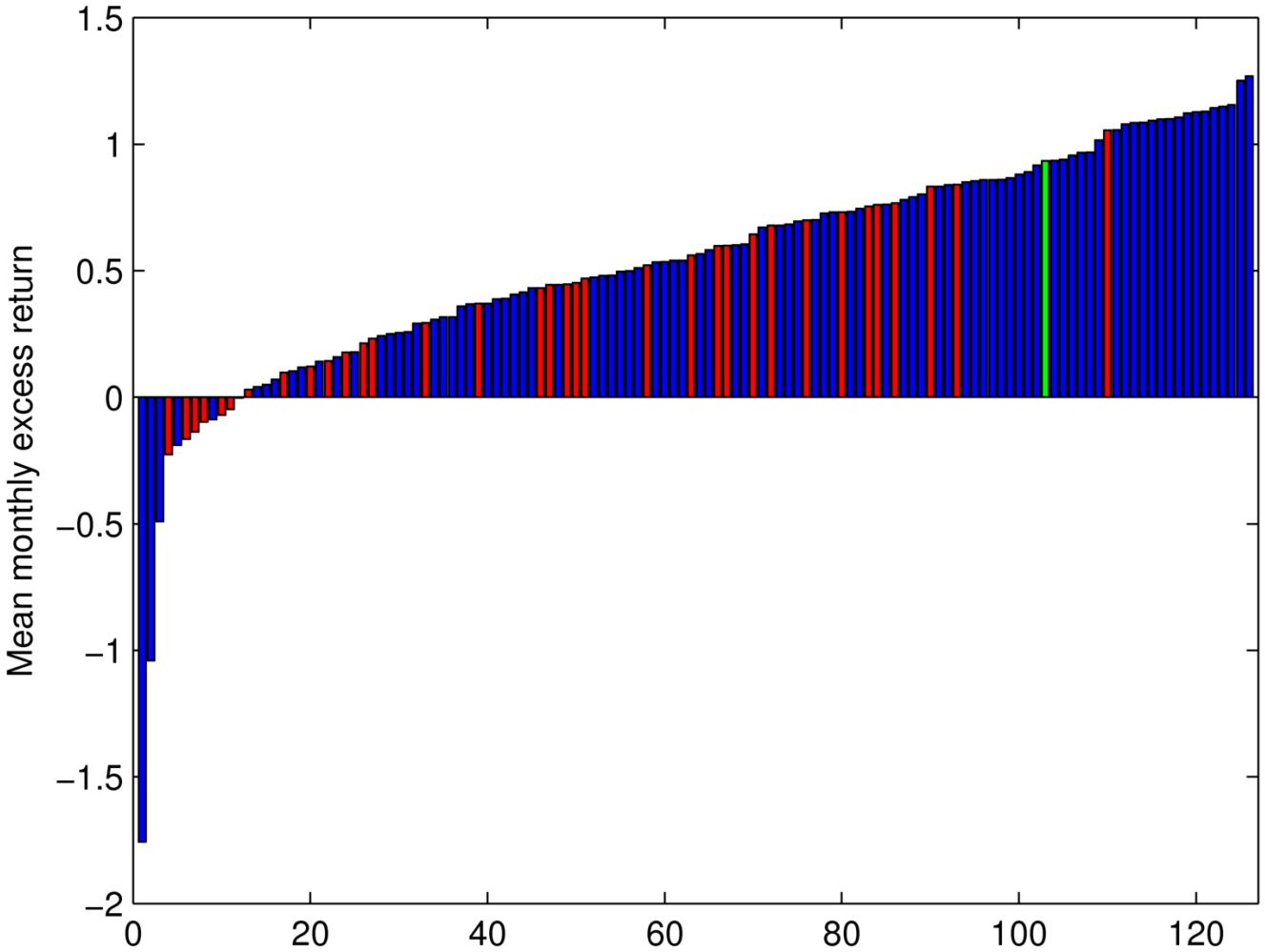
# SINGLE FEATURE PERFORMANCE



# SINGLE FEATURE PERFORMANCE: SH ($d \leq 1$)



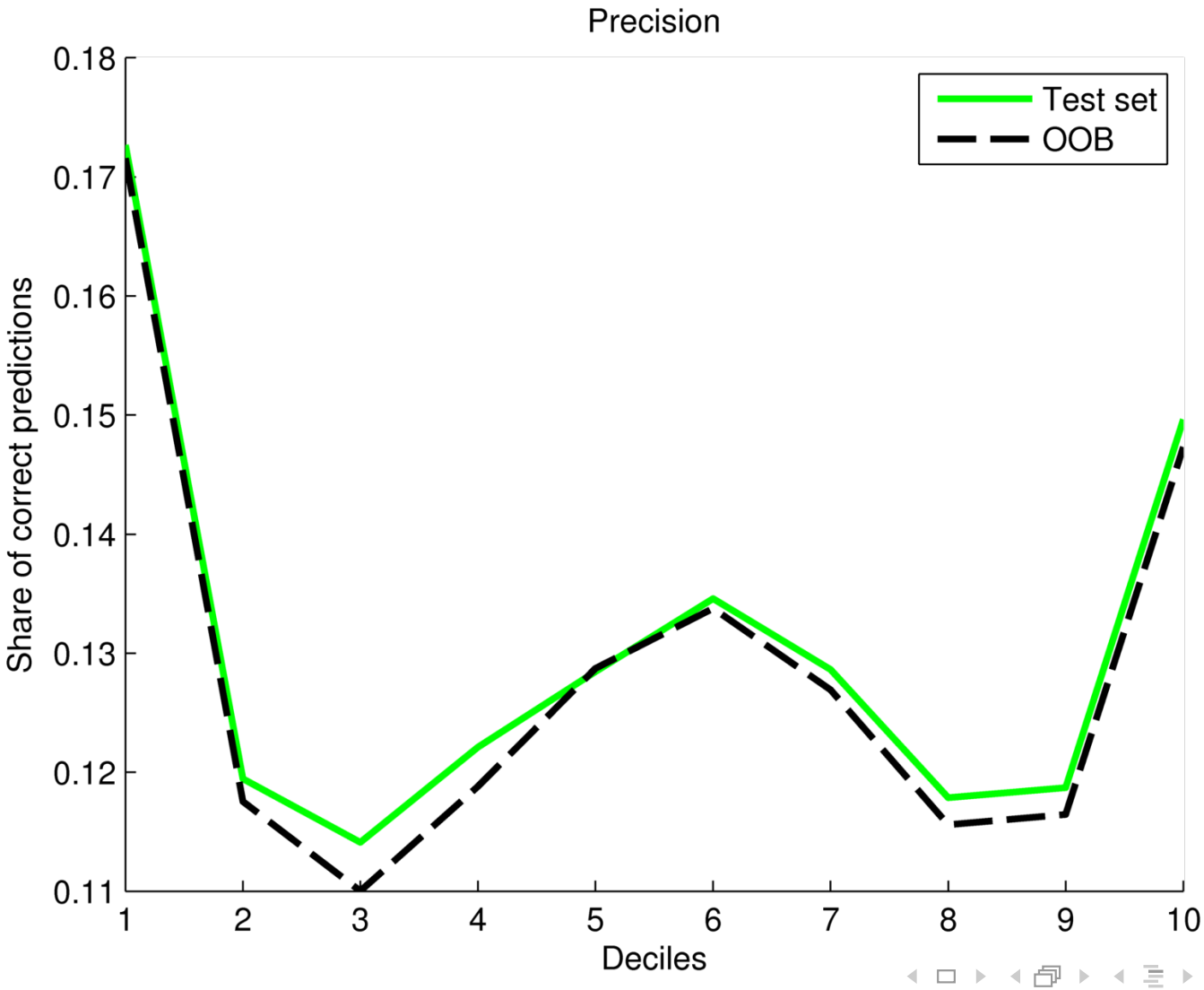
# SINGLE FEATURE PERFORMANCE: MH ($d \in \{5, 6\}$)



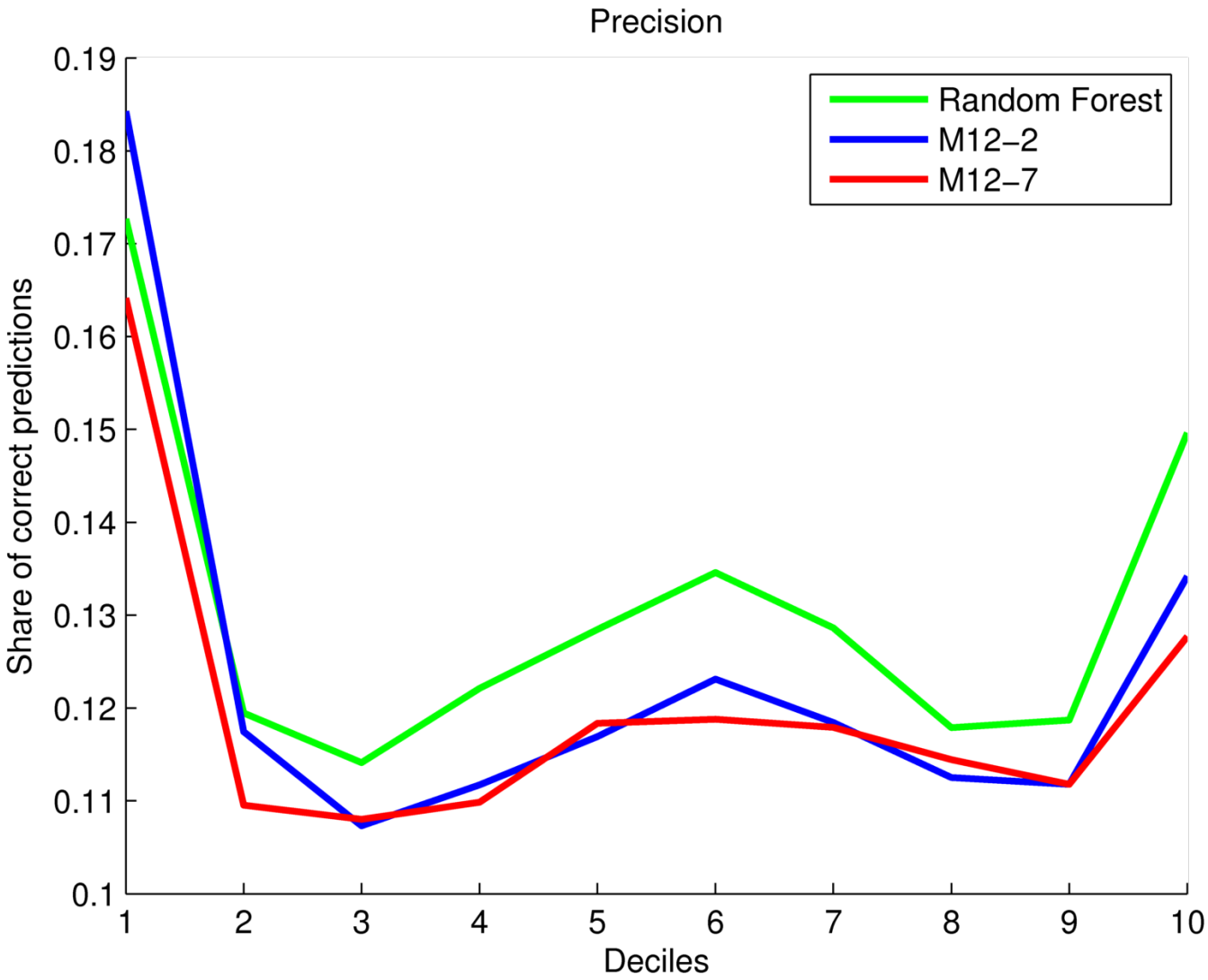
# IMPLEMENTATION

- Construct 126 predictors $M(d, l)$
- Compute decile sorting for each predictor and one-month future return
- Tuning parameters (tree depth, number of randomly chosen predictors,...) can be cross-validated (not done yet)
- One RF run takes 6-7 hours with parallel processing

# RANDOM FOREST PRECISION: OOB AND HOLD-OUT SAMPLE



# RANDOM FOREST: PRECISION



# RANDOM FOREST: FEATURE IMPORTANCE

| $d$ (lag) | $l$ (length) | Relative Importance |
|---|---|---|
| 0 | 1 | 1 |
| 0 | 2 | .660 |
| 2 | 1 | .655 |
| 1 | 1 | .595 |
| 0 | 3 | .555 |
| 2 | 3 | .555 |
| 1 | 3 | .542 |
| 1 | 2 | .541 |
| 4 | 2 | .527 |
| 3 | 1 | .502 |

# RANDOM FOREST: FEATURE IMPORTANCE

| $d$ (lag) | $l$ (length) | Relative Importance |
|---|---|---|
| 0 | 1 | 1 |
| 0 | 2 | .660 |
| 2 | 1 | .655 |
| 1 | 1 | .595 |
| 0 | 3 | .555 |
| 2 | 3 | .555 |
| 1 | 3 | .542 |
| 1 | 2 | .541 |
| 4 | 2 | .527 |
| 3 | 1 | .502 |

# RANDOM FOREST: FEATURE IMPORTANCE

| Most important | | Least important | |
|---|---|---|---|
| $d$ (lag) | $l$ (length) | $d$ (lag) | $l$ (length) |
| 0 | 1 | 5 | 12 |
| 0 | 2 | 2 | 12 |
| 2 | 1 | 3 | 11 |
| 1 | 1 | 2 | 17 |
| 0 | 3 | 0 | 17 |
| 2 | 3 | 0 | 14 |
| 1 | 3 | 2 | 15 |
| 1 | 2 | 3 | 16 |
| 4 | 2 | 2 | 13 |
| 3 | 1 | 3 | 12 |

# RANDOM FOREST: FEATURE IMPORTANCE

| Most important | | Least important | |
|---|---|---|---|
| $d$ (lag) | $l$ (length) | $d$ (lag) | $l$ (length) |
| 0 | 1 | 5 | 12 |
| 0 | 2 | 2 | 12 |
| 2 | 1 | 3 | 11 |
| 1 | 1 | 2 | 17 |
| 0 | 3 | 0 | 17 |
| 2 | 3 | 0 | 14 |
| 1 | 3 | 2 | 15 |
| 1 | 2 | 3 | 16 |
| 4 | 2 | 2 | 13 |
| 3 | 1 | 3 | 12 |

Brief look at simple strategy return:

- Buy highest predicted quantile
- Sell lowest predicted quantile

# RANDOM FOREST: FACTOR LOADINGS

Strategy: Long highest predicted quantile, short lowest predicted quantile

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| Intercept | .97 | 1.14 | 1.26 | .76 |
| | (5.60) | (5.56) | (5.86) | (3.67) |
| MKT | | -.18 | -.10 | .01 |
| | | (-1.80) | (-1.21) | (.18) |
| SMB | | | -.18 | -.16 |
| | | | (-1.58) | (-1.54) |
| HML | | | -.36 | -.14 |
| | | | (-3.70) | (-1.55) |
| UMD | | | | .47 |
| | | | | (8.02) |

# RANDOM FOREST: FACTOR LOADINGS

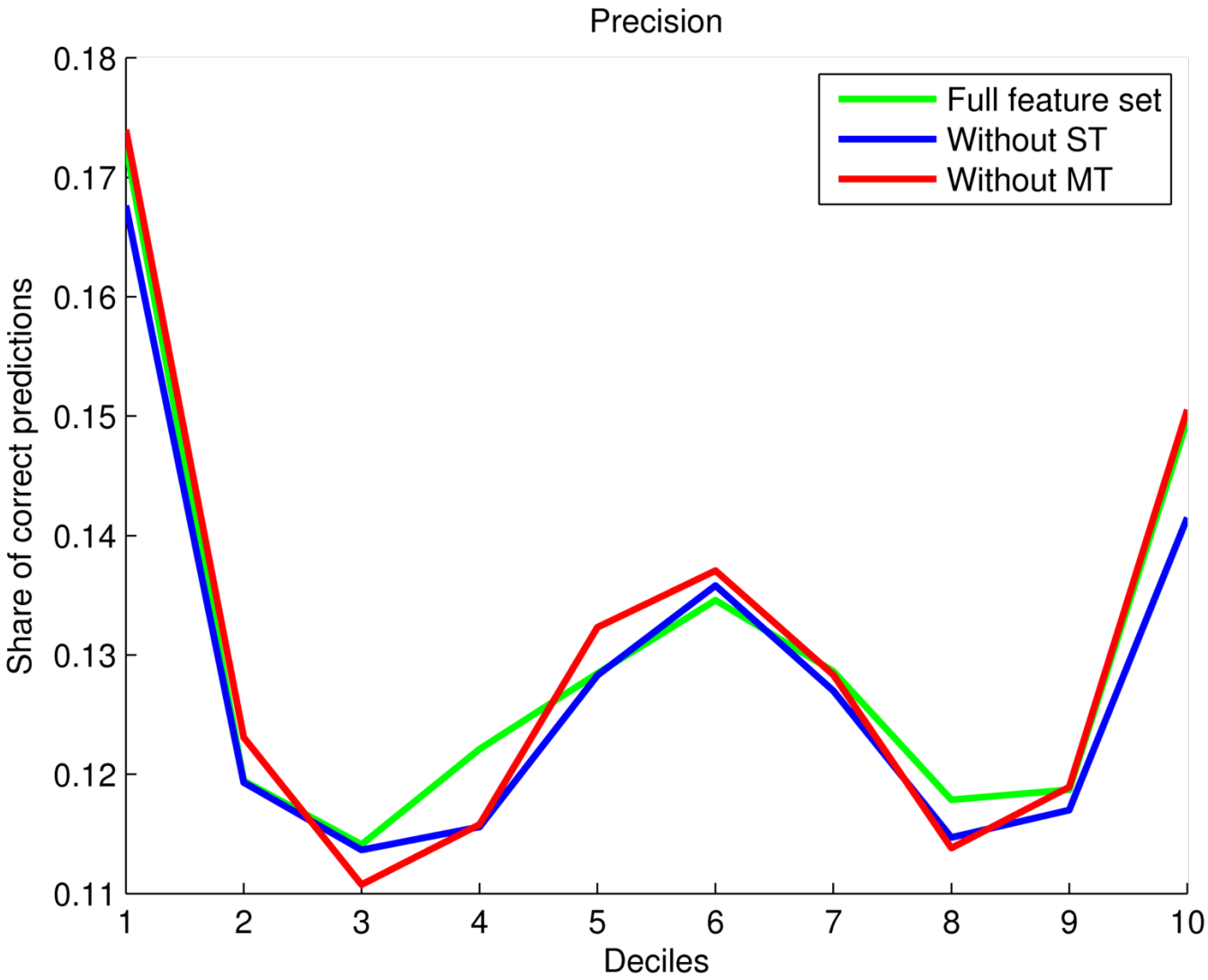
Strategy: Long highest predicted quantile, short lowest predicted quantile

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| Intercept | .97 | 1.14 | 1.26 | .76 |
| | (5.60) | (5.56) | (5.86) | (3.67) |
| MKT | | -.18 | -.10 | .01 |
| | | (-1.80) | (-1.21) | (.18) |
| SMB | | | -.18 | -.16 |
| | | | (-1.58) | (-1.54) |
| HML | | | -.36 | -.14 |
| | | | (-3.70) | (-1.55) |
| UMD | | | | .47 |
| | | | | (8.02) |

# EXCLUDING FEATURES

- What is the relevance of whole classes of features?
- Idea: Does exclusion of one group result in performance deterioration in the prediction task?
- Precision might even be higher if focus on more relevant features
- Define short-horizon features as $d \leq 1$, medium-horizon as $d \geq 5$

# RF WITH RESTRICTED FEATURE SETS



# MOST IMPORTANT FEATURES

| | $\{X : d \leq 4\}$ | | $\{X : d \geq 2\}$ | |
|---|---|---|---|---|
| Rank | $d$ (lag) | $l$ (length) | $d$ (lag) | $l$ (length) |
| 1 | 0 | 1 | 2 | 1 |
| 2 | 0 | 2 | 2 | 2 |
| 3 | 1 | 1 | 5 | 1 |
| 4 | 1 | 2 | 2 | 3 |
| 5 | 0 | 3 | 6 | 1 |

# MOST IMPORTANT FEATURES

| | $\{X : d \leq 4\}$ | | $\{X : d \geq 2\}$ | |
|---|---|---|---|---|
| Rank | $d$ (lag) | $l$ (length) | $d$ (lag) | $l$ (length) |
| 1 | 0 | 1 | 2 | 1 |
| 2 | 0 | 2 | 2 | 2 |
| 3 | 1 | 1 | 5 | 1 |
| 4 | 1 | 2 | 2 | 3 |
| 5 | 0 | 3 | 6 | 1 |

# FACTOR LOADINGS

| | $\{X : d \leq 4\}$ | | | | $\{X : d \geq 2\}$ | | | |
|---|---|---|---|---|---|---|---|---|
| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) |
| Intercept | 1.34 | 1.46 | 1.59 | 1.08 | .82 | .97 | 1.08 | .48 |
| | (6.42) | (6.49) | (6.94) | (5.04) | (4.55) | (5.00) | (5.18) | (2.38) |
| MKT | | -.15 | -.10 | -.02 | | -.16 | -.09 | .02 |
| | | -(1.92) | -(1.55) | -(.34) | | -(2.44) | -(1.35) | (.28) |
| SMB | | | -.09 | -.05 | | | -.21 | -.20 |
| | | | -(1.03) | -(.67) | | | -(2.30) | -(2.73) |
| HML | | | -.38 | -.19 | | | -.28 | -.05 |
| | | | -(3.71) | -(2.11) | | | -(2.71) | -(.46) |
| UMD | | | | .47 | | | | .57 |
| | | | | (8.17) | | | | (10.59) |

# FACTOR LOADINGS

| | $\{X : d \leq 4\}$ | | | | $\{X : d \geq 2\}$ | | | |
|---|---|---|---|---|---|---|---|---|
| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) |
| Intercept | 1.34 | 1.46 | 1.59 | 1.08 | .82 | .97 | 1.08 | .48 |
| | (6.42) | (6.49) | (6.94) | (5.04) | (4.55) | (5.00) | (5.18) | (2.38) |
| MKT | | -.15 | -.10 | -.02 | | -.16 | -.09 | .02 |
| | | -(1.92) | -(1.55) | -(.34) | | -(2.44) | -(1.35) | (.28) |
| SMB | | | -.09 | -.05 | | | -.21 | -.20 |
| | | | -(1.03) | -(.67) | | | -(2.30) | -(2.73) |
| HML | | | -.38 | -.19 | | | -.28 | -.05 |
| | | | -(3.71) | -(2.11) | | | -(2.71) | -(.46) |
| UMD | | | | .47 | | | | .57 |
| | | | | (8.17) | | | | (10.59) |

# SUMMARY: EXCLUDING FEATURES

- Eyeballing suggests that precision suffered by excluding SH features, and did not change excluding MH features
- Return regressions point in the same direction
- Need to test this formally
- Additional test: Predict SH features with MH features (and vice versa) and use fitted values to predict future returns (flavor of Granger causality test)

# SUMMARY & OUTLOOK

# SUMMARY

- Preliminary framework to better understand cross-section of stock returns
- Controls for correlated predictors and specification search
- Momentum results
  - SH returns can be learned from the data, MH returns not
  - Strategy return is not subsumed by UMD
  - Momentum might be a more complex phenomenon than can be captured by any one specific past return

# OUTLOOK

- We are currently expending the feature set (volatility, skewness, firm characteristics)
- Add value-weighted strategies
- Transfer learning: Can our findings be transferred to other markets? Or is each market characterized by a different interplay of past return variables?
- Hope is to speed-up the process of scientific discovery in this area and better understand the current state of knowledge